E-FILED on 8/8/2013

IN THE UNITED STATES DISTRICT COURT

FOR THE NORTHERN DISTRICT OF CALIFORNIA

SAN JOSE DIVISION

NAZOMI COMMUNICATIONS, INC.,

Plaintiff,

v.

SAMSUNG TELECOMMUNICATIONS, INC., *et al.*,

Defendants.

No. C-10-05545 RMW

ORDER GRANTING DEFENDANTS' MOTION FOR SUMMARY JUDGMENT OF NONINFRINGEMENT AND DENYING DEFENDANTS' MOTION FOR SUMMARY JUDGMENT OF INVALIDITY OF U.S. PATENT NO. 6,338,160

[Re Docket Nos. 245, 246]

On October 22, 2010, defendants filed concurrent motions for summary judgment of noninfringement and summary judgment of invalidity of U.S. Patent No. 6,338,160 (the '160 patent). On November 28, 2012, the court held a hearing to consider defendants' motions and to construe the disputed terms of the '160 patent. The claim construction order, which this court is issuing concurrently with this summary judgment order, provides the basis for the noninfringement and invalidity determinations contained herein. Based on the court's claim construction, the papers submitted by the parties and the arguments of counsel, and for the reasons set forth below, the court:

(1) GRANTS defendants' motion for summary judgment of noninfringement; and (2) DENIES defendants' motion for summary judgment of invalidity.

I. BACKGROUND

A. The Technology

The invention of the '160 patent is a method for resolving references to a constant pool at runtime in the programming language Java. *See* '160 patent col.1 ll.30-59; *see* Claim Construction Order, Dkt. No. 278 (providing a more detailed description of the claimed invention). According to the '160 patent, at runtime, an "invoke instruction" references the constant pool and "cause[s] the [unresolved] reference to be resolved." '160 patent col.1 ll.54-59. The claimed constant pool includes a "data resolution field" (also called a "resolution data field") that indicates whether the reference to an object has been resolved. *Id.* col.2 ll.19-39. For example, a "0" in the data resolution field would indicate that the reference has not been resolved and "cause[] the system to jump to a location . . . in the native instruction region [where] the resolve instructions" are located; whereas a "1" in the field would indicate that the reference is resolved and cause the system to jump to the "native instruction for the invoke instruction." *Id.* col.8 ll.32-52; *see also id.* figs.9A and 9B. The invention also includes an "indication field," which, before resolution, contains an indication of the reference that needs to be resolved. *Id.* col.8 ll.32-52, col.10 ll.3-4, and fig.9A. Once the reference is resolved, the indication field is updated to "indicate[] the location of the loaded object." *Id.* col.8 ll.40-48 and fig.9B.

B. The Accused Products

The accused products are devices running Google's Android operating system, which uses the "Dalvik Virtual Machine." "Similar to how the Java Virtual Machine uses Java bytecodes to allow for portability across different platforms (*e.g.*, Intel x86, ARM), the Dalvik Virtual Machine uses bytecodes referred to as Dalvik bytecodes." Decl. of Marc E. Levitt, Ph.D. ¶ 9, Dkt. No. 251 ("Levitt Decl."); *see also* Decl. of David I. August, Ph.D. ¶ 23, Dkt. No. 246-35 ("August Decl."). In contrast to Java's stack-based instructions, "Dalvik bytecodes are register-based instructions." August Decl. ¶ 23. Applications for the Android operating system are written using Java. Java files are compiled in ".class" or ".jar" files ("Java class files"), which cannot be run on the Dalvik Virtual

Machine. Nevertheless, the Dalvik Virtual Machine can run Java programs by using a tool called "dx" to convert the Java class files into an executable Dalvik ".dex" file. Unlike the Java Virtual Machine, which allocates separate constant pools for each class file, in the Dalvik Virtual Machine, the "dx" tool combines the constants from each of the Java class file constant pools into a single .dex file constant pool. In the .dex file, any repetitive constants, i.e., constants that existed in more than one Java class file, are eliminated.

At runtime, when the .dex file is loading, the Dalvik Virtual Machine "allocates data structures *associated with* the '.dex' file" (the structures are defined in a file called "DvmDex.h"). Levitt Decl. ¶ 15 (emphasis added). "In particular, the DvmDex.h file defines the DvmDex structure which includes the pResMethods structure identified in Nazomi's infringement contentions." *Id.* ¶ 17; *see also* Nazomi's Infringement Contentions, Ex. B 43-44, Dkt. No. 245-14 ("Infringement Contentions") ("[I]n the [accused devices], the invoke_virtual instruction references an entry in pResMethods in a constant pool."). Dr. Levitt characterizes the DvmDex structure in the Dalvik Virtual Machine as the "constant pool . . . described in the '160 patent because it references an object that may require runtime resolution." Levitt Decl. ¶ 20.

When the pResMethods structure is allocated, prior to the resolution of any reference, each entry in the pResMethods structure is initialized to 0. Levitt Decl. ¶ 18; August Decl. ¶ 30. After the resolution step, the pResMethods entry corresponding to that particular reference or "method" is updated to contain a valid address corresponding to the location of the resolved method in memory. Levitt Decl. ¶ 26; August Decl. ¶¶ 31, 32. The parties' experts disagree as to whether the data in the pResMethods entries (the 0 or the memory address for the resolved reference) corresponds to the "indication of a reference that may need resolution" and/or "data resolution field" limitations in the asserted claims.

Dr. Levitt concludes that the data entry in pResMethod is both a "data resolution field" and "an indication of a reference that may need resolution." With respect to the "data resolution field" limitation, Dr. Levitt concludes that, because the pResMethods entry "is compared to 0 to determine if the entry . . . is resolved," it "is used to determine whether to perform a resolving step." Levitt Decl. ¶ 23. Dr. Levitt also concludes that the data entry in "pResMethods provides an indication of a reference that may need resolution because when a bytecode references [the] entry . . . , the value of 0 will indicate that resolution needs to be performed." *Id.* ¶ 19.

In contrast, Dr. August concludes that the data entry in pResMethods is neither a "data resolution field," as construed by defendants, nor an "indication of a reference that may need resolution." Dr. August concludes that "pResMethods does not have a 'resolution data field'" because "pResMethods has only one field per entry [and t]herefore, . . . cannot have one field . . . which is separate from another . . . ." August Decl. ¶ 34. Dr. August also concludes that the data entry in pResMethods never contains an indication of a reference that many need resolution because the value in the entry is either 0 (before resolution) or a valid memory address (after resolution), neither of which indicate the name of the reference to be resolved. *Id.* ¶¶ 32, 33.

C. Procedural Posture

Nazomi contends that the accused products infringe claims 11, 15, 18, and 21 of the '160 patent. Claim 11 is the only independent claim, from which the others depend. Defendants move for summary judgment of noninfringement under their proposed constructions and summary judgment of invalidity under Nazomi's proposed constructions. The court has construed the terms, adopting defendants' construction, in substantive part, of two key terms ("constant pool" and "indication of a reference that may need resolution"), which simplifies the summary judgment analysis because: (1) Nazomi concedes that there is no direct infringement under these constructions; and (2) defendants only argue invalidity under Nazomi's proposed constructions for "constant pool" and "indication of a reference that may need resolution."

Specifically, the court has construed the disputed terms as follows:

Disputed Term	Construction
"constant pool"	a data structure attached to a single loaded class that encodes all the names that can be used by any method in the loaded class
"constant pool entry" / "entry in a constant pool"	an entry within the constant pool [as defined above]

"instruction" "executing an instruction"	either a stack-based instruction that is to be translated into a register-based instruction, or a register-based instruction that is input to the CPU pipeline executing an instruction [as defined above]
"an indication of a reference that may need resolution"	an identification of a location (e.g., an address) within the constant pool that stores the name, or "label," of a reference that needs resolution
"resolution data field" / "data resolution field"	a data field within the constant pool entry that contains data indicating whether a reference has been resolved

Claim Construction Order 16.

II. NONINFRINGEMENT

Summary judgment is appropriate when there is no genuine issue of material fact such that the moving party is entitled to judgment as a matter of law. Fed. R. Civ. P. 56(a); *Celotex Corp. v. Catrett*, 447 U.S. 317, 322-23 (1986). Where a defendant seeks summary judgment of non-infringement, "nothing more is required than the filing of a . . . motion stating that the patentee had no evidence of infringement and pointing to the specific ways in which accused [products] did not meet the claim limitations." *Exigent Tech. v. Atrana Solutions, Inc.*, 442 F.3d 1301, 1309 (Fed. Cir. 2006). The burden of production then shifts to the patentee to "identify genuine issues that preclude summary judgment." *Optivus Tech., Inc. v. Ion Beam Applications S.A.*, 469 F.3d 978, 990 (Fed. Cir. 2006). Infringement, both literal and under the doctrine of equivalents, is a question of fact, and thus "is amenable to summary judgment where, *inter alia*, no reasonable fact finder could find infringement." *Ethicon Endo-Surgery, Inc. v. U.S. Surgical Corp.*, 149 F.3d 1309, 1315 (Fed. Cir. 1998). If the parties do not dispute any relevant facts regarding the accused product, "but disagree over possible claim interpretations, the question of literal infringement collapses into claim construction and is amenable to summary judgment." *Gen. Mills, Inc. v. Hunt-Wesson, Inc.*, 103 F.3d 978, 983 (Fed. Cir. 1997). Nevertheless, as with all summary judgment motions, the court must view all evidence in the light most favorable to the non-moving party and draw all reasonable

United States District Court
For the Northern District of California

inferences in its favor. *IMS Tech., Inc. v. Haas Automation, Inc.*, 206 F.3d 1422, 1429 (Fed. Cir. 2000).

In order to establish a prima facie case of direct infringement, Nazomi must show that the moving defendants make, use, sell, offer to sell, or import a product that infringes at least one asserted claim. *See* 35 U.S.C. § 271(a). An infringement analysis entails two steps: (1) determining the meaning and scope of the patent claims; and (2) comparing the construed claims to the devices accused of infringing. *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 976 (Fed. Cir. 1995), *aff'd*, 517 U.S. 370 (1996). To prove infringement, Nazomi must show that the accused products "meet[] each claim limitation either literally or under the doctrine of equivalents." *Seachange Int'l, Inc. v. C-COR, Inc.*, 413 F.3d 1361, 1377 (Fed. Cir. 2005). Literal infringement requires that the accused device contain each of the claim elements and recited limitations of the claims at issue. *See Signtech USA, Ltd. v. Vutek, Inc.*, 174 F.3d 1352, 1358 (Fed. Cir. 1999).

An accused device may also infringe under the doctrine of equivalents. An accused product that does not literally infringe may still infringe under the doctrine of equivalents if "the accused product or process contain elements identical or equivalent to each claimed element of the patented invention." *Warner-Jenkinson Co. v. Hilton Davis Chem. Co.*, 520 U.S. 17, 21 (1997). The premise of the doctrine of equivalents is "language's inability to capture the essence of innovation" and its goal is to prevent fraud on the patent through an overly literal reading of the claims. *Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co.*, 535 U.S. 722, 734 (U.S. 2002); *Graver Tank & Mfg. Co. v. Linde Air Products Co.*, 339 U.S. 605, 606 (U.S. 1950). A court can find infringement even in the absence of literal infringement if each element of the accused device does the same work in substantially the same way, and accomplishes substantially the same result as disclosed in the patent. *Warner-Jenkinson Co. v. Hilton Davis Chem. Co.*, 520 U.S. 17, 39 (1997); *Graver Tank & Mfg. Co. v. Linde Air Products Co.*, 339 U.S. 605, 608 (1950); *Lockheed Martin Corp. v. Space Sys./Loral, Inc.*, 324 F.3d 1308, 1320 (Fed. Cir. 2003). A doctrine of equivalents analysis is conducted on a limitation by limitation basis (the "all elements rule"). *Warner-Jenkinson*, 520 U.S. at 39-40. Under the "all elements rule" a patentee may not use the doctrine of equivalents when its application would "vitiate a claim limitation." *Abbot Laboratories v. Andrx Pharmaceuticals*, 473 F.3d 1196, 1212

(Fed. Cir. 2007). Subject matter cannot be included within the scope of a patent under the doctrine of equivalents if it is inconsistent with the language of the claim. *See SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1347 (Fed. Cir. 2001)*; Ethicon Endo-Surgery, Inc. v. U.S. Surgical Corp.*, 149 F.3d 1309, 1317 (Fed. Cir. 1998).

Nazomi contends that the accused products directly infringe the asserted claims based on its proposed claim constructions of the disputed terms. Nazomi alleges infringement under the doctrine of equivalents with respect to only the constant pool limitation. Defendants argue that they do not infringe the asserted claims because Nazomi's infringement contentions are based on incorrect claim constructions, and the accused product does not contain each of the following disputed claim limitations under the correct claim constructions. The court address each claim limitation in turn.

A. The "constant pool" limitation

1. The "constant pool" in the accused device

As an initial matter, the parties disagree as to which structure in the accused device actually corresponds to the constant pool limitation in claim 11. Defendants characterize the "constant pool" in the accused product as the pResMethods data structure. In contrast, Nazomi contends that the "constant pool" is the entire DvmDex structure, which, according to Nazomi and Dr. Levitt, "associates" with the .dex file at runtime, and includes the pResMethods structure. *See* Levitt Decl. ¶¶ 15, 16.

Nazomi's infringement contentions identify "the invoke_virtual instruction [in the Dalvik Virtual Machine that] references an entry *in pResMethods <u>in</u> a constant pool*" as the structure that meets the "instruction that references an entry in a constant pool" limitation in claim 11. Infringement Contentions at 43-44 (emphasis added). The infringement contentions, thus, indicate that pResMethods itself is not the accused "constant pool." For the purposes of summary judgment, the court views the facts in a light most favorable to Nazomi, and considers the entire DvmDex structure, including pResMethods, to be the accused "constant pool."

2. Direct infringement

Defendants' opening brief argues that pResMethods, standing alone, does not meet the constant pool limitation of claim 11. In their reply, defendants argue in the alternative that the

DvmDex stucture, including the pResMethods structure, does not meet the constant pool limitation. Defendants primary arguments with respect to pResMethods, however, are equally applicable with respect to the DvmDex structure including pResMethods. First, defendants argue that pResMethods (and similarly the DvmDex structure including pResMethods) is not a constant pool because it "corresponds to an *entire Dex file* when an Android program is loaded into memory . . . [and u]nlike the constant pools in Java, Dalvik does not create a separate pResMethods array for *each class*." Defs.' Br. in Support of Mot. for Summ. J. at 11, Dkt. No. 245 ("Defendants' Br.") (emphases added). Second, defendants argue that, unlike Java, "pResMethods itself [or the DvmDex structure including pResMethods] does not contain 'all the names that can be used by any method in the loaded class.'" *Id.* at 12. Finally, in its reply brief, defendants argue that the accused product does not meet the claim limitation requiring "an instruction that references an entry *in* a constant pool" because the entry in pResMethods is not "in" the .dex file constant pool. *See* Levitt Decl. ¶ 16 (characterizing the pResMethods structure as becoming "associated with" the .dex file at runtime, but not as being "in" the .dex file constant pool).

Nazomi argues that the accused devices include a constant pool under its broad claim construction proposal: "a data structure that includes at least one constant pool entry." The court adopted defendants' proposed construction (replacing the word "table" with "data structure"), construing the term "constant pool" to mean: "a data structure attached to a single loaded class that encodes all the names that can be used by any method in the loaded class." Nazomi does not argue direct infringement under defendants' proposed construction.

3. Infringement under the Doctrine of Equivalents

In the alternative, Nazomi argues under defendants' proposed construction that the constant pool in the accused products is equivalent to the "constant pool" in the asserted claims, and thus, defendants infringe under the doctrine of equivalents. Defendants do not respond to Nazomi's doctrine of equivalents argument, presumably, as argued by defendants in related Case No. C-10-04686, because Nazomi did not properly assert infringement under the doctrine of equivalents in the infringement contentions. *See* Infringement Contentions p. 3, ¶ E (including only a "catch-all," boilerplate reservation under the doctrine of equivalents). The Patent Local Rules require a limitation-by-limitation analysis for infringement. *See* Patent Local R. 3-1(d) and (e). A boilerplate reservation is inadequate, and courts have frequently dismissed claims under the doctrine of equivalents based upon boilerplate language in their infringement contentions. *See Rambus Inc. v. Hynix Semiconductor Inc.*, C-05-00334 RMW, 2008 WL 5411564, *3 (N.D. Cal. Dec. 29, 2008) (finding in the alternative that it could grant summary judgment for failure to comply with the patent rules' "limitation-by-limitation" requirement). Nevertheless, because defendants did not raise any objection in this case to Nazomi's doctrine of equivalents argument, and because the court adopted a claim construction that renders only the doctrine of equivalents relevant, the court consider's Nazomi's argument in the alternative. *See* Patent Local R. 3-6 (allowing amendments to infringement contention by order of the court upon a timely showing of good cause, including "a claim construction by the [c]ourt different from that proposed by the party seeking amendment").

Nazomi's equivalence argument is based on Dr. Levitt's declaration. Dr. Levitt states that the "shared constant pool in Dalvik performs the *same function* as the individual constant pools in the [Java] '.class' or '.jar' files." Levitt Decl. ¶ 13 (emphasis added). According to Dr. Levitt, both (1) "the ability to transform .class/.jar files to .dex files via the dx tool in a straight forward manner" and (2) "[t]he correct operation of such .class/.jar files in the Dalvik VM" evidence that "the operation of the constant pool in the '.dex' file operates in substantially the same way to yield substantially the same result." *Id.* ¶ 14. Dr. Levitt states that, if the accused constant pool did not operate in substantially the same way to yield substantially the same result, then "the application level operation of the Java based program would be substantially different between Java VM and Dalvik VM versions," which is not the case. *Id.* Although Dr. Levitt states that the Dalvik Virtual Machine accomplishes an "equivalent" result, he offers minimal to no explanation of any "substantially similar *way*" in which the Dalvik Virtual Machine achieves that "equivalent result." The mere existence of equivalent results is not sufficient, by itself, to establish a substantially similar "way." The only statement Dr. Levitt offers that can reasonably be interpreted as alleging a substantially similar "way" is his statement that "each entry in the Java .class file can be reconstructed and found in the .dex file constant pool." *Id.* ¶ 14. Even viewing this statement in a light most favorable to Nazomi, however, the mere statement that same constant pool entries exist in the Java and Dalvik does create a question of fact as to whether the accused system operates in substantially the same way as the Java constant pool. Accordingly, the court grants defendants' motion for summary judgment of noninfringement based on the "constant pool" limitation.

B. The "indication of a reference that may need resolution" and "resolution data field" limitations

The court addresses both of these claim limitations together because Nazomi alleges that the same data entry in the pResMethods structure in the accused products meets *both* of these limitations.

Defendants argue that pResMethods: (1) does not include an indication of a reference that may need resolution because the data entry in pResMethods does not store "an indication of the *name* of a reference" to be resolved; and (2) does not include a "data resolution field" because the "data resolution field" must be separate from the indication of a reference that requires resolution, and the entry in pResMethods includes only one field.

Nazomi counters that the accused products: (1) contain an "indication of a reference that may need resolution" under its proposed claim construction, which requires the "indication of a reference that may need resolution" to indicate "*whether* a reference has been resolved," for example, with the entry of a "0" or "1" in the field, *see* Levitt Decl. ¶ 19 ("pResMethods provides an indication of a reference that may need resolution because when a bytecode references an entry in pResMethods, *the value of 0 will indicate that the resolution needs to be performed*." (emphasis added)); and (2) contain a "data resolution field" because there is no physical separation requirement for the "data resolution field" and the "indication of a reference that may need resolution."

The court construed the term "indication of a reference that may need resolution" to mean: "an identification of a location (e.g., an address) within the constant pool that stores the name, or "label," of a reference that needs resolution," and the term "resolution data field" to mean: "a data field within the constant pool entry that contains data indicating whether a reference has been resolved."

Under the court's constructions, the data in the pResMethods entries (i.e., the 0 or the valid memory address for the resolved reference) does not contain "an indication of a reference that may need resolution" because it does not identify "a location (e.g., an address) within the constant pool that stores the name, or 'label,' of a reference that needs resolution." Dr. Levitt based his infringement conclusion to the contrary on Nazomi's erroneous claim construction, wherein he presumed that the "indication of a reference" indicated "*whether* a reference had been resolved." For example, Dr. Levitt concluded that the data entry in "pResMethods provides an indication of a reference that may need resolution because when a bytecode references [the] entry . . . , the value of 0 will indicate that resolution needs to be performed." *Id.* ¶ 19. The court rejected such a construction of this term, and Nazomi makes no infringement argument in the alternative. Accordingly, the court grants defendants' motion for summary judgment of noninfringement of the asserted claims because the accused products do not contain an "indication of a reference that may need resolution."

There is an issue of fact, however, as to whether data entry in pResMethods meets the "data resolution field" claim limitation because it could be "a data field within the constant pool entry that contains data indicating whether a reference has been resolved." (A factual dispute remains as to whether the data field is *within* the constant pool.). Dr. August's conclusion that "pResMethods does not have a 'resolution data field' / 'data resolution field'" was based on the fact that "pResMethods has only one field per entry." August Decl. ¶ 34. The court declined to construe "resolution data field" to require two separate fields, and thus, the data field in the pResMethods entry *could* be a "resolution data field" if it is within a constant pool entry. Whether or not this is true, however, will not affect the court's noninfringement determination because the only field in the pResMethods entry does not contain "an indication of a reference that may need resolution," and thus this limitation is not met by the accused products. Accordingly, the court also grants defendants' motion for summary judgment of noninfringement based on the "indication of a reference that may need resolution" claim limitation.

C. The "instruction" limitation

The court rejected defendants' narrow proposed construction for "instruction," and declined to limit an "instruction" to a stack-based instruction. Defendants do not argue noninfringement under the court's construction, and this limitation is not a basis for the court's noninfringement determination on summary judgment.

III. INVALIDITY

Defendants' motion for summary judgment of invalidity under 35 U.S.C. § 102(a) is premised on the claim limitations at issue being construed according to Nazomi's proposed construction. The court, however, adopted, in substantive part, the majority of *defendants*' proposed constructions. Defendants do not argue that each and every claim limitation, as now construed, is anticipated. Accordingly, in light of the claim construction order in this case, the court denies defendants' motion for summary judgment of invalidity based on Nazomi's proposed claim constructions.

V. ORDER

For the foregoing reasons, the court GRANTS defendants' motion for summary judgment of noninfringement of the '160 patent, and DENIES defendants' motion for summary judgment of invalidity of the '160 patent.

DATED: August 8, 2013

Ronald M. Whyte

RONALD M. WHYTE
United States District Judge